This listing of claims will replace all prior versions, and listings, of the claims in this application:

Listing of Claims

Claim 1 (currently amended): A computer-implemented method, comprising the steps of:

establishing a plurality of virtual machines;

establishing a plurality of partitions of processor time;

assigning each virtual machine of the plurality of virtual machines to a partition of the plurality of partitions;

running, on a single processor, each virtual machine during its assigned partition; and

determining whether a virtual machine ~~has any action to perform during its assigned partition and~~ will [[thus]] be inactive during its assigned partition.

Claim 2 (canceled)

Claim 3 (canceled)

Claim 4 (original): The method of claim 1, wherein said assigning step takes into account results of prior determining steps in making assignments of virtual machines to partitions.

Claim 5 (canceled)

Claim 6 (original): The method of claim 1, further comprising the step of placing the single processor into a reduced power mode during a partition assigned to a virtual machine that has been determined to be inactive by said determining step.

Claim 7 (canceled)

Claim 8 (canceled)

Claim 9 (original): The method of claim 6, wherein the reduced power mode is terminated at the end of the partition assigned to the inactive virtual machine.

Claim 10 (original): The method of claim 1, further comprising the step of reassigning, to another virtual machine, a partition previously assigned to a virtual machine that has been determined to be inactive by said determining step.

Claim 11 (original): The method according to claim 10, wherein said reassigning step assigns a partition associated with an inactive virtual machine to the virtual machine assigned to the next partition.

Claim 12 (original): The method according to claim 10, wherein said reassigning step assigns a partition associated with an inactive virtual machine to the next occurring partition that has been assigned to a virtual machine determined not to be inactive.

Claim 13 (currently amended): A computing apparatus, comprising:

a memory component storing code establishing a plurality of virtual machines, establishing a plurality of partitions of processor time, assigning each virtual machine of the plurality of virtual machines to a specific partition of the plurality of partitions, and determining whether a virtual machine ~~has any action to perform during its assigned partition and~~ will [[thus]] be inactive during its assigned partition;

a processor, coupled with said memory component, said processor being capable of running each virtual machine during its assigned partition and of running code stored on said memory component; and

wherein said memory component also stores code placing said processor into a lower power mode during a partition assigned to an inactive virtual machine.

Claim 14 (original): The apparatus according to claim 13, wherein said processor comprises an embedded, low power processor.

Claim 15 (canceled)

Claim 16 (canceled)

Claim 17 (canceled)

Claim 18 (canceled)

Claim 19 (currently amended): A computing apparatus, comprising:

a memory component storing code establishing a plurality of virtual machines, establishing a plurality of partitions of processor time, assigning each virtual machine of the plurality of virtual machines to a specific partition of the plurality of partitions, and determining whether a virtual machine will be inactive such that it will not be performing an action during its assigned partition;

a processor, coupled with said memory component, to run each virtual machine during its assigned partition and to run code stored on said memory component; and

wherein said memory component also stores code activating a subsequent virtual machine during a partition assigned to an inactive virtual machine.

Claim 20 (currently amended): A computing apparatus, comprising:

means for storing code establishing a plurality of virtual machines, establishing a plurality of partitions of processor time, assigning each virtual machine of the plurality of virtual machines to a specific partition of the plurality of partitions, and determining whether a virtual machine ~~has any action to perform~~ will be inactive during its assigned partition;

means for processing, coupled with said means for storing, said means for processing running each virtual machine during its assigned partition and running code stored on said means for storing; and

wherein said means for storing also stores code placing said means for processing into a reduced power mode for the duration of a partition that has been determined to have an inactive virtual machine.

Claim 21 (currently amended): A computing apparatus, comprising:

means for storing code establishing a plurality of virtual machines, establishing a plurality of partitions of processor time, assigning each virtual machine of the plurality of virtual machines to a specific partition of the plurality of partitions, and determining whether a virtual machine ~~has any action to perform~~ will be inactive during its assigned partition;

means for processing, coupled with said means for storing, said means for processing running each virtual machine during its assigned partition and running code stored on said means for storing; and

wherein said means for storing also stores code reassigning, to another virtual machine, a partition previously assigned to a virtual machine that has been determined to be inactive.

Claim 22 (currently amended): A computer-readable storage medium, comprising:

a computer-executable code for establishing a plurality of virtual machines, establishing a plurality of partitions of processor time, assigning each virtual machine of the plurality of virtual machines to a specific partition of the plurality of partitions, determining whether a virtual machine will be inactive such that it will not be performing an action during its assigned partition, and for activating a subsequently scheduled virtual machine for the duration of a partition that has been determined to have an inactive virtual machine.

Claim 23 (currently amended): A computer-readable storage medium, comprising:

a computer-executable code for establishing a plurality of virtual machines, establishing a plurality of partitions of processor time, assigning each virtual machine of the plurality of virtual machines to a specific partition of the plurality of partitions, determining whether a virtual machine will be inactive such that it will not be performing an action during its assigned partition, and for activating a reduced power mode for the duration of a partition that has been determined to have an inactive virtual machine.

Claim 24 (currently amended): A computer-readable storage medium, comprising:

a computer-executable code to establish a virtual machine schedule for activating a plurality of virtual machines, to determine whether a scheduled virtual machine will be inactive such that it will not be performing an action during its scheduled activation time, and to initiate a reduced power mode for the duration of an inactive virtual machine's scheduled activation time.

Claim 25 (currently amended): A computer-readable storage medium, comprising:

a computer-executable code to establish a virtual machine schedule for activating a plurality of virtual machines, to determine whether a scheduled virtual machine will be inactive such that it will not be performing an action during its scheduled activation time, and to initiate reassignment, to another virtual machine, of a ~~partition~~ scheduled activation time previously assigned to a virtual machine that has been determined to be inactive.

Claim 26 (currently amended): A computer-implemented method, comprising the steps of:

establishing a virtual machine schedule for activating, on a single processor, a plurality of virtual machines;

determining whether a scheduled virtual machine will be inactive such that it will not be performing an action during its scheduled activation time; and

initiating processor entry of a reduced power mode for the duration of an inactive virtual machine's scheduled activation time.

Claim 27 (currently amended): A computer-implemented method, comprising the steps of:

establishing a virtual machine schedule for activating, on a single processor, a plurality of virtual machines;

determining whether a scheduled virtual machine will be inactive such that it will not be performing an action during its scheduled activation time; and

initiating reassignment of an inactive virtual machine's scheduled activation time to a virtual machine determined to be active.

Claim 28 (currently amended): A computer-implemented method, comprising the steps of:

establishing a plurality of JAVA virtual machines;

establishing a plurality of partitions of processor time;

assigning each JAVA virtual machine of the plurality of JAVA virtual machines to a partition of the plurality of partitions;

running, on a single embedded low power JAVA processor, each JAVA virtual machine during its assigned partition;

determining whether a JAVA virtual machine to be run ~~has any action to perform during its assigned partition and~~ will [[thus]] be inactive during its assigned partition;

placing the single embedded low power JAVA processor into a reduced power mode during a partition assigned to the JAVA virtual machine that has been determined to be inactive by said determining step; and

exiting the reduced power mode at the end of the partition assigned to the inactive JAVA virtual machine and placing the single embedded low power JAVA processor into a higher power mode.

Claim 29 (new) The method of claim 1, wherein said determining step determines whether a virtual machine will be inactive by checking a status field for a halt code.

Claim 30 (new) The method of claim 29, wherein the halt code indicates the occurrence of an error.

Claim 31 (new) The method of claim 29, wherein the halt code indicates that a virtual machine has not loaded.

Claim 32 (new) The method of claim 29, wherein the halt code indicates that a virtual machine execution error has occurred.

Claim 33 (new) The method of claim 29, wherein the halt code indicates that a processor specific error has occurred.

Claim 34 (new) The method of claim 29, wherein the halt code indicates that a partition time limit has been exceeded.

Claim 35 (new) The method of claim 29, wherein the halt code indicates that a partition space restriction has been exceeded.

Claim 36 (new) The method of claim 29, wherein the halt code indicates that a power down handler did not complete successfully.

Claim 37 (new) The method of claim 29, wherein the halt code indicates an invalid initialized data block.